Машина Тьюринга. Тезис Чёрча-Тьюринга

**Машина Тьюринга** (МТ) — абстрактный конечный исполнитель (абстрактная вычислительная машина состояний) предложенный Аланом Тьюрингом в 1936 году.

В начале 30-х годов двадцатого века одной из актуальных проблем математики являлась “проблема вычислимости”. Общую идею такой проблемы можно выразить следующим образом: возьмем некую функцию или неравенство, например (1), можно ли однозначно доказать, что это неравенство имеет решение? Более того, можно ли вывести такой общий метод решения, который позволит для любого произвольного выражения определить решаемо оно или нет.

An+bn=Rn (1)

Данная проблема относится к фундаментальным проблемам математики и привлекло множество энтузиастов к её решению. В частности к схожим результатам в одно и тоже время пришли такие исследователи как Тьюринг, Пост и Чёрч.

Работы Поста и Тьюринга во многом схожи, так как оба исследователя решили разработать абстрактного исполнителя, который позволил бы сформулировать понятия алгоритма. Само понятие алгоритма известно еще с работ арабского исследователя [Аль Хорезми](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D1%8C-%D0%A5%D0%BE%D1%80%D0%B5%D0%B7%D0%BC%D0%B8), однако изначально алгоритмы были только правилами выполнения математических действий. Теперь же нам нужны а четко зафиксированные правила для выполнения любых произвольных операций. Для четкого их фиксирования необходим конкретный исполнитель. Именно Тьюринга (или, например, Поста) будет таким исполнителем. Причём этот исполнитель является, по сути, математической абстракцией.

Более поздние исследователи, полагаясь на [оригинальную работу Тьюринга](https://ru.wikipedia.org/wiki/%D0%A2%D1%8C%D1%8E%D1%80%D0%B8%D0%BD%D0%B3,_%D0%90%D0%BB%D0%B0%D0%BD) разработали множество других исполнителей и сегодня мы имеем богатый их набор, однако в нашей работе мы будем использовать тот вариант, который наиболее близок к оригинальной идее.

Прежде чем продолжить рассказ о проблеме вычислимости и её выводах, необходимо разобрать принципы работы машины Тьюринга (МТ) и решения задач с её использованием.

# Устройство и работа машины Тьюринга

В своей оригинальной работе Тьюринг сформулировал два понятия: Машина Тьюринга и Универсальная Машины Тьюринга. Мы начнем разбор с обычной машины, а позднее покажем, как будет выглядеть универсальная.

Основная идея МТ заключается в том, что у нас есть память, разделенная на отдельные ячейки, каждая из которых может хранить одно значение. Такая память потенциально может быть бесконечной (рис.1 ).

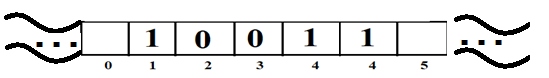


Рис 1. Иллюстрация машины Тьюринга

В дальнейшем мы не будем рисовать бесконечность влево и вправо, и для простоты учета действий будем подписывать номера ячеек так, как это показано на рисунке 1.

Каждая ячейка на ленте может хранить любое возможное значение, однако перечень или диапазон таких значений необходимо сформулировать заранее, и они будут составлять *Алфавит машины.* Оригинальная машина Тьюринга использует алфавит из трёх символов: 1, 0 и «пусто». Это означает, что в каждой ячейке может быть записан один из этих символов. Этот подход будет использоваться и нами, с его помощью мы будем выполнять двоичную арифметику.

Кроме того, машина включает читающую головку, которая может по требованию перемещаться по ленте вправо и влево (по одной ячейке за раз) и указывает на одну из ячеек (рис 2). Таким образом, в любой случайно выбранный момент времени мы можем считать значение текущей ячейки, а можем поменять его на другое.

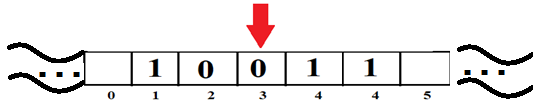


Рис 2. Иллюстрация машины Тьюринга

Сегодня данное устройство легко может быть реализовано физически, но для наших целей подойдет и абстрактная модель. Кроме того, МТ включает таблицу, описывающую действия, которые будут выполняться над лентой и правила их применения. Перечень соответвий действий правилам мы будем называть состоянием. Традиционно каждое состояние описывают буквой q с номером. Таким образом формируется внутренний алфавит машины, состоящий из набора состояний

Q = {q0, q1,q2…qs}.

Здесь Q - это полный набор всех состояний, q0 – начальное состояние, qs – конечное, все остальные промежуточные. Каждое состояние можно описать набором параметров, пример которых приведен в таблице 1.

Таблица 1. Пример машины Тьюринга

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условие | Действие | Переход |
| q0 | пусто | Сдвиг вправо | q0 |

Параметры представляют собой следующие характеристики:

* Название: на самом деле они могут быть любые, например - “Первое”, “Начальное”, “Отличное”, “1” и так далее. Запись с буквой qи последующим за ни номером является просто традиционной.
* Условие: проверяется значение ячейки, на которую указывает читающая головка. Если значение ячейки совпадает со значением условия, то выполняется то, что записано в Действии и Переходе
* Действие: выполняется описанная команда, или их набор. Действие может быть любым – Сдвиг вправо, сдвиг влево, стереть значение, заменить значение на 0 и тому подобные.
* Переход: предписывает машине следующее состояние.

Машина Тьюринга, представленная в таблице 1, состоит из одного состояния и выполняется до тех пор, пока не встретит первый непустой символ. Как только она встретит непустой символ, она не сможет двигаться дальше, так как нет такого состояния, которое описывало бы поведение в случае, если читающая головка указывает на ячейку с непустым символом. Для иллюстрации работы данной машины возьмем ленту, показанную на рисунке 3, и начнем пошаговое исполнение.

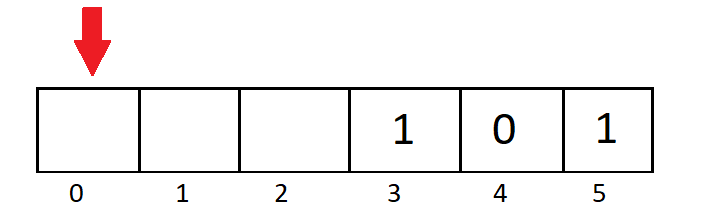


Рис 3.Начало работы машины из таблицы 1

Шаг 1. В начале работы машины Тьюринга из таблицы 1, читающая головка указывает на ячейку с номером 0, в которой ничего нет, а состояние машины соответствует состоянию q0. Так как значение текущей ячейки “пусто”, что соответствует условию, мы передвигаем головку на одну ячейку вправо, и переходим к состоянию q0, таким образом лента принимает вид, показанный на рисунке 4.

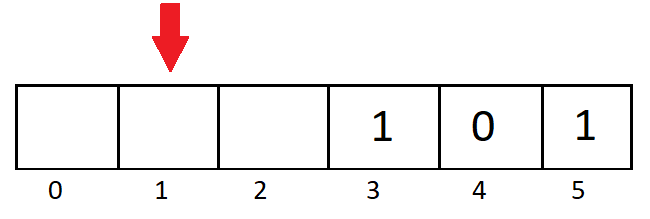


Рис 4.Шаг 1 машины таблицы 1

Шаг 2, состояние q0. Значение ячейки на которую указывает головка “пусто”, это соответствует условию для состояния q0, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 5.

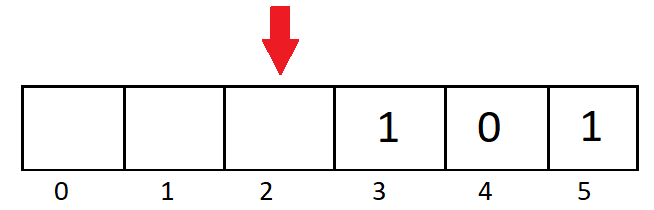


Рис 5. Шаг 2 машины таблицы 1

Шаг 3, состояние q0. Значение ячейки на которую указывает головка “пусто”, это соответствует условию для состояния q0, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 6.

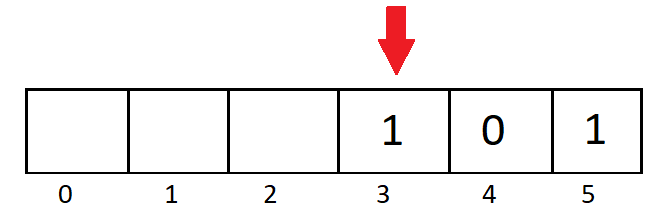


Рис 6.Шаг 3 машины таблицы 1

Шаг 4, состояние q0. Значение ячейки, на которую указывает головка “1”. Для состояния q0 не описано поведение в этом случае, следовательно, машина остановит свое выполнение с ошибкой.

Работа машины Тьюринга завершена с ошибкой, так как неизвестно что делать дальше. В случае, если бы лента была бесконечная и пустая, машина тоже работала бы бесконечно. Для того, чтобы сделать машину интереснее, зададим для состояния q0 несколько правил, как в таблице 2.

Таблица 2. Пример машины Тьюринга со стиранием

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условие | Действие | Переход |
| q0 | Пусто | Сдвиг вправо | q0 |
| q0 | 1 | Стереть значение  Сдвиг вправо | q0 |

В таблице 2 мы видим программу, стирающую все единицы с ленты памяти. Она, состоит из одного состояния и выполняется до тех пор, пока не встретит ноль. Как только она встретит ноль, она не сможет двигаться дальше, так как нет такого состояния, которое описывало бы поведение в случае, если читающая головка указывает на ячейку с другим непустым символом. Для иллюстрации работы данной машины возьмем ленту, показанную на рисунке 7.

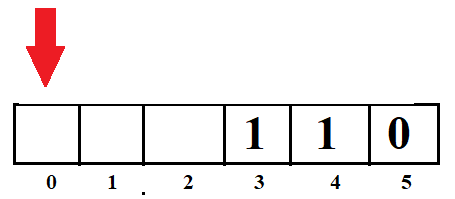


Рис 7.Начало работы машины таблицы 2

Шаг 1. В начале работы машины Тьюринга из таблицы 1, читающая головка указывает на ячейку с номером 0, в которой ничего нет, а состояние машины соответствует состоянию q0. Так как значение текущей ячейки “пусто”, что соответствует условию, мы передвигаем головку на одну ячейку вправо, и переходим к состоянию q0, таким образом лента принимает вид, показанный на рисунке 8.

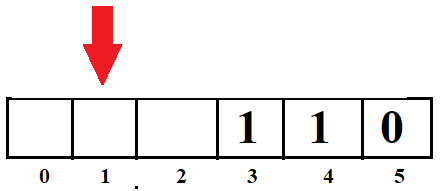


Рис 8.Шаг 1 машины таблицы 2

Шаг 2, состояние q0. Значение ячейки на которую указывает головка “пусто”, это соответствует условию для состояния q0, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 9.

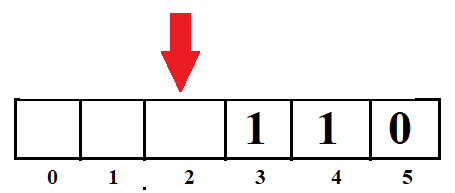


Рис 9.Шаг 2 машины таблицы 2

Шаг 3, состояние q0. Значение ячейки, на которую указывает головка “пусто”, это соответствует условию для состояния q0, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 10.

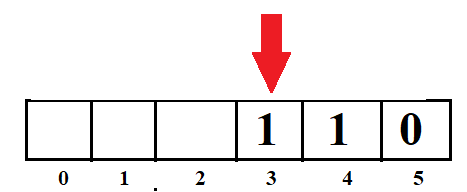


Рис 10.Шаг 3 машины таблицы 2

Шаг 4, состояние q0. Значение ячейки, на которую указывает головка “1“, это соответствует второму условию для состояния q0, следовательно, стираем имеющееся значение и выполняем сдвиг на одну ячейку вправо и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 11.

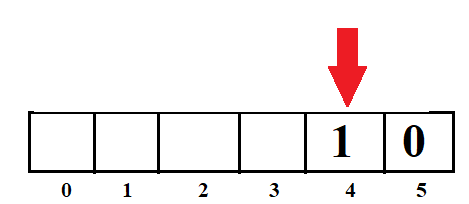


Рис 11.Шаг 4 машины таблицы 2

Шаг 5, состояние q0. Значение ячейки, на которую указывает головка “1“, это соответствует второму условию для состояния q0, следовательно, стираем имеющееся значение и выполняем сдвиг на одну ячейку вправо и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 12.

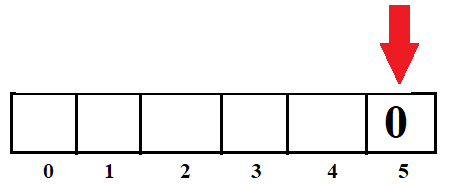


Рис 12.Шаг 5 машины таблицы 2

Шаг 6, состояние q0. Значение ячейки, на которую указывает головка “0”. Для состояния q0 не описано поведение в этом случае, следовательно, машина остановит свое выполнение с ошибкой.

Примечательно, что технически данная программа не останавливается никогда, так как она не имеет правил остановки. Нет такого условия, встретив которое, программа завершит выполнение. Однако для машины Тьюринга это не является проблемой. Более того, она даже не является *циклической*.

Циклической будем называть такую машину, которая бесконечно повторяет одни и те же конфигурации машины. Под конфигурацией будем понимать сочетание: значений ленты, положения читающей головки и текущего состояния. Остальные машины назовем ациклическими. Например, если считывающая головка “застревает” между конфигурациями, бесконечно переходя из одной ячейки в другую и назад, то такая машина будет являться циклической (см. рис. 13 и табл. 3).

Таблица 3. Пример циклической МТ

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условие | Действие | Переход |
| … | … | … | … |
| qn | 1 | Заменить на 0  Сдвиг вправо | qv |
| qn | 0 | Заменить на 1  Сдвиг вправо | qv |
| qv | 1 | Заменить на 0  Сдвиг влево | qn |
| qv | 0 | Заменить на 1  Сдвиг влево | qn |
| … | … | … | … |

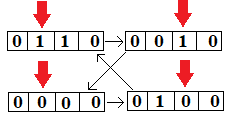


Рис 13. Циклическая МТ

В таблице 3 мы видим пример циклической машины Тьюринга, а точнее, фрагмент её набора правил, из-за которой она является таковой. Попав в ячейку с «1» головка выполняет правила состояния qn, в том числе и переход к состоянию qv. Второе состояние возвращает головку на предыдущее место и переходит к первому состоянию. Таким образом, мы видим, что данная машина бесконечно перемещается между четырьмя конфигурациями, и таким образом, попала в бесполезный бесконечный цикл. Таким машины мы будем называть *циклическими.*

Со временем правилом хорошего тона стало считаться необходимость задавать точку остановки, то есть такое правило, при котором МТ прекращает свою работу. Модифицируем машину Тьюринга, представленную в таблице 2 таким образом, чтобы она имела точку остановки, то есть состояние, при котором выполнение машины заканчивается. Результатом будет машина, представленная в таблице 4.

Таблица 4. МТ с состоянием завершения работы.

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условие | Действие | Переход |
| q0 | Пусто | Сдвиг вправо | q0 |
| q0 | 1 | Стереть значение | q0 |
| q0 | 0 |  | qs |
| qs |  | Стоп |  |

В таблице 4 можно увидеть, что для состояния q0 добавилось новое правило, которое говорит о том, что при значении в ячейке равным нулю, необходимо перейти в состояние qs. Состояние qs в свою очередь выполняет только одной действие – остановку машины. Продолжим выполнение машины, показанной в таблице 2, с того шага на котором была ошибка (т.е. шестой шаг), заменив этот шаг на новый вариант.

Шаг 6, состояние q0. Значение ячейки, на которую указывает головка “0“, это соответствует третьему условию для состояния q0, следовательно, переходим к состоянию qs.

Шаг 7, состояние qs. Остановка машины. Работа завершена без ошибок.

Теперь попробуем разработать МТ, у которой будет три состояния (см. Таблицу 5)

Таблица 5. Инвертирование знаков

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условие | Действие | Переход |
| q0 | Пусто | Сдвиг вправо | q0 |
| q0 | Не(Пусто) |  | q1 |
| q1 | 1 | Записать 0  Сдвинуться вправо | q1 |
| q1 | 0 | Записать 1  Сдвинуться вправо | q1 |
| q1 | Пусто |  | qs |
| qs |  | Стоп |  |

Машина Тьюринга, описанная в таблице 5, выполняет инвертирование знаков некоей двоичной записи. Данная машина представлена состояниями, выполняющими следующие действия:

1. Состояние q0 необходимо, чтобы найти ближайшую непустую запись справа. Оно начинает работать с некоей случайно выставленной ячейки и сдвигается вправо пока не найдет первую непустую ячейку.
2. Состояние q1 инвертирует значения ячеек до тех пор, пока не встретит первую непустую ячейку
3. Состояние qs останавливает работу.

Обратим внимание на некоторые особенности:

1. В блоке “действие” одному правилу мы приписали несколько действий
2. Во втором правиле q0 выполняется проверка на отрицание значения

Заметим, что если на ленте окажется, записано несколько двоичных последовательностей записанных через пробел, то инвертирована будет только первая встретившаяся. В качестве дополнительного задания читатель может попытаться так модифицировать машину из таблицы 4, что бы она работала бесконечно, но меняла все имеющиеся на ленте последовательности.

Разберем пошагово, как работает МТ в таблице 5.Для иллюстрации работы данной машины возьмем ленту, показанную на рисунке 14.

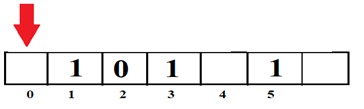


Рис 14.Начало работы машины таблицы 5

Шаг 1. В начале работы машины Тьюринга из таблицы 5, читающая головка указывает на ячейку с номером 0, в которой ничего нет, а состояние машины соответствует состоянию q0. Так как значение текущей ячейки “пусто”, что соответствует условию, мы передвигаем головку на одну ячейку вправо, и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 15.

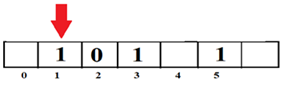


Рис 15.Шаг 1 машины таблицы 5

Шаг 2, состояние q0. Значение ячейки, на которую указывает головка “1“, это соответствует условию для состояния q0 - “не пусто”, следовательно, переходим к состоянию q1, причем никаких действий не выполняется, так как они не описаны. Лента остается в том же состоянии, как и на рисунке 15.

Шаг 3, состояние q1. Значение ячейки, на которую указывает головка “1“, это соответствует первому условию для состояния q1, следовательно, заменяем «1» на «0», выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 16.

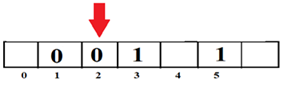


Рис 16.Шаг 3 машины таблицы 5

Шаг 4, состояние q1. Значение ячейки, на которую указывает головка “0“, это соответствует второму условию для состояния q1, следовательно, заменяем «0» на «1», выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 17.

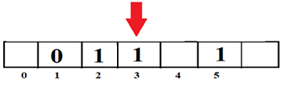


Рис 17.Шаг 4 машины таблицы 5

Шаг 5, состояние q1. Значение ячейки, на которую указывает головка “1“, это соответствует первому условию для состояния q1, следовательно, заменяем «1» на «0», выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 18.

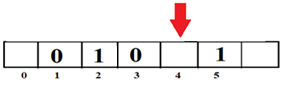


Рис 18.Шаг 5 машины таблицы 5

Шаг 6, состояние q1. Значение ячейки, на которую указывает головка “пусто”, это соответствует третьему условию для состояния q1, следовательно, переходим к состоянию qs. Сдвиг не выполняется, головка остаётся на месте.

Шаг 7, состояние qs. Завершение работы без ошибок.

Пока что, все построенные нами машины были необходимы только для иллюстрации, и не выполняли особенно полезных действий. Для того, чтобы продемонстрировать настоящие возможности машины, нам нужен более очевидный пример использования, скажем, добавление единицы к некоему случайному двоичному числу, записанному на ленте. Очевидно, что если можно выполнить сложение, то можно выполнить вообще все что угодно.

Машина Тьюринга необходимая для этого описана в таблице 6.

Таблица 6. Прибавление единицы к числу

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условие | Действие | Переход |
| q0 | Пусто | Сдвиг вправо | q0 |
| q0 | Не пусто | Сдвиг вправо | q1 |
| q1 | Не пусто | Сдвиг вправо | q1 |
| q1 | Пусто | Сдвиг влево | q2 |
| q2 | 1 | Записать 0  Сдвиг влево | q2 |
| q2 | 0 | Записать 1 | qs |
| qs |  | Стоп |  |

Машина Тьюринга, описанная в таблице 6, представлена состояниями, выполняющими следующие действия:

1. Состояние q0 снова используется для того чтобы найти ближайшую непустую запись справа.
2. Состояние q1 начинает работу с первого знака нашего числа и ищет его конец, сдвигаясь вправо.
3. Состояние q2 идет по разрядам числа справа налево и заменяет все единицы на нули до тех пор, пока не встретит первый ноль, который будет заменен на единицу. Легко проверить, что это поведение соответствует логике двоичного сложения.
4. Состояние qs останавливает работу.

Разберем пошагово, как работает МТ в таблице 6.Для иллюстрации работы данной машины возьмем ленту, показанную на рисунке 19.

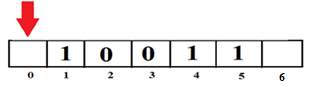


Рис 19.Начало работы машины таблицы 6

Шаг 1. В начале работы машины Тьюринга из таблицы 5, читающая головка указывает на ячейку с номером 0, в которой ничего нет, а состояние машины соответствует состоянию q0. Так как значение текущей ячейки “пусто”, что соответствует условию, мы передвигаем головку на одну ячейку вправо, и переходим к состоянию q0, таким образом, лента принимает вид, показанный на рисунке 20.

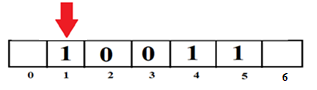


Рис 20.Шаг 1 машины таблицы 6

Шаг 2, состояние q0. Значение ячейки, на которую указывает головка “1“, это соответствует условию для состояния q0 - “не пусто”, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 21.

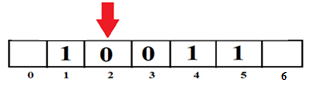


Рис 21.Шаг 2 машины таблицы 6

Шаг 3, состояние q1. Значение ячейки, на которую указывает головка “0“, это соответствует условию для состояния q1 - “не пусто”, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 22.

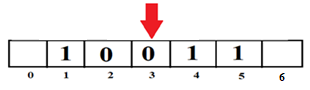


Рис 22.Шаг 3 машины таблицы 6

Шаг 4, состояние q1. Значение ячейки, на которую указывает головка “0“, это соответствует условию для состояния q1 - “не пусто”, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 23.

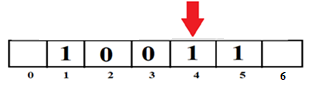


Рис 23.Шаг 4 машины таблицы 6

Шаг 5, состояние q1. Значение ячейки, на которую указывает головка “1“, это соответствует условию для состояния q1 - “не пусто”, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 24.

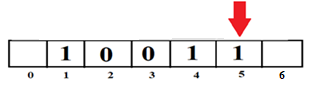


Рис 24.Шаг 5 машины таблицы 6

Шаг 6, состояние q1. Значение ячейки, на которую указывает головка “1“, это соответствует условию для состояния q1 - “не пусто”, следовательно, выполняем сдвиг на одну ячейку вправо и переходим к состоянию q1, таким образом, лента принимает вид, показанный на рисунке 25.

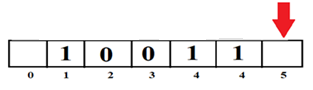


Рис 25.Шаг 6 машины таблицы 6

Шаг 7, состояние q1. Значение ячейки, на которую указывает головка “пусто“, это соответствует условию для состояния q1, следовательно, выполняем сдвиг на одну ячейку влево и переходим к состоянию q2, таким образом, лента принимает вид, показанный на рисунке 26.

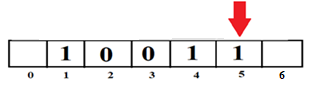


Рис 26.Шаг 7 машины таблицы 6

Шаг 8, состояние q2. Значение ячейки, на которую указывает головка “1“, это соответствует условию для состояния q2, следовательно, заменяем «1» на «0», выполняем сдвиг на одну ячейку влево и переходим к состоянию q2, таким образом, лента принимает вид, показанный на рисунке 27.

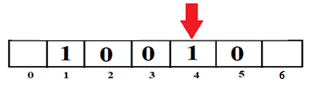


Рис 27.Шаг 8 машины таблицы 6

Шаг 9, состояние q2. Значение ячейки, на которую указывает головка “1“, это соответствует условию для состояния q2, следовательно, заменяем «1» на «0», выполняем сдвиг на одну ячейку влево и переходим к состоянию q2, таким образом, лента принимает вид, показанный на рисунке 28.

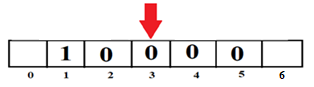


Рис 28.Шаг 9 машины таблицы 6

Шаг 10, состояние q2. Значение ячейки, на которую указывает головка “0“, это соответствует условию для состояния q2, следовательно, заменяем «0» на «1» и переходим к состоянию qs, таким образом, лента принимает вид, показанный на рисунке 29.

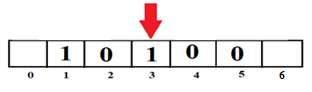


Рис 29.Шаг 10 машины таблицы 6

Шаг 11, состояние qs. Завершение работы без ошибок.

Данная функция просматривает всё двоичное число и ищет последнюю цифру. Если она равна 0, то заменяем на 1. Иначе заменяем 1 на 0 и возвращаемся назад, повторяя эту операцию до тех пор, пока не встретится 0. После этого выполнение прекращается.

Задача добавления единицы к произвольному двоичному числу решена. Исходя из этого, можно продолжить развитие идеи, реализовать сложение двух двоичных числе, далее остальные математические операции и так до тех пор, пока не получится полноценный компьютер. Однако описанной машине не хватает универсальности.

# 2. Универсальная машина Тьюринга

Основной идеей универсальности выступает возможность имитации поведения одной машины Тьюринга другой машиной. Для этого нам необходимо модифицировать ленту так, как показано на рисунке 30.



Рис 30. Схема универсальной машины Тьюринга

Любую таблицу состояний (например те, что разобраны выше) мы можем записать во внешний алфавит А. Таким образом, начальные ячейки памяти занимает программа. Словесное описание алгоритма может быть неточным, поэтому описание алгоритма работы МТ необходимо сделать с помощью машины Тьюринга, т.е. поставить задачу построения МТ, реализующей алгоритм воспроизведения работы исполнительного механизма МТ.

Исходя из этого строится первый тезис Тьюринга: любой произвольный исполнитель может имитировать другой исполнитель и следовательно они взаимозаменяемы.

Построенную универсальную машину Тьюринг использовал для доказательства главной идеи: невозможно однозначно доказать, что какая-либо функция однозначна верна или нет (т.е. вычислима), что и являлось целью построения машины.

Доказательство (в очень общих чертах) базируется на следующих утверждениях:

1. Если какая-либо частная машина оказывает циклической, следовательно, функция недоказуема
2. Что бы доказать, что частная машина циклическая нужна универсальная машина, которая проведет анализ работы частной машины и докажет, что она циклическая
3. Для того что бы провести анализ нужно выполнить частную машину и, следовательно, также попасть в бесконечный цикл так ничего и не доказав
4. Поскольку все машины взаимозаменяемы – значит невозможно создать такой способ, который скажет циклическая ли конкретная машина, а значит и разрешима ли некая функция.

# Тезис Чёрча - Тьюринга

Работа Тьюринга (а также некоторых других исследователей) позволила нам сформулировать несколько основных терминов используемых в современных информационных технологиях.

1. Универсальная машина Тьюринга – это аналог современно компьютера и языка программирования для него одновременно
2. Лента – это оперативная память компьютера, в которой записываются одновременно данные и команды для исполнения
3. Частная машина Тьюринга – это аналог функции или подпрограммы, которая написана на понятном исполнителю языке.
4. Алгоритм – это последовательность шагов, которая может исполняться машиной Тьюринга.

Поскольку одновременно несколько исследователей достигли схожих результатов, то сегодня выделяют общий тезис **Чёрча – Тьюринга,** который гласит: *всякая вычислимая функция вычислима машиной Тьюринга.*

Черч разработал концепцию лямбда вычислений о которой мы поговорим позднее.

Для современного IT эта информация результирует в виде нескольких интересных выводов.

Во-первых, любой алгоритмический язык взаимозаменяем с другим языком, если все они **Тьюринг полные**. Под полнотой по Тьюрингу будем понимать возможность написать имитацию машины Тьюринга на каком-либо языке. Практически для любого языка находится энтузиаст доказавший его Тьюринг полноту, и следовательно одно и то же приложение всегда может быть написано на любом желаемом языке. Разница между разными языками программирования заключается только в удобстве использования их для решения конкретных задач.

Во-вторых, говорят о **Тьюринг-трясине**. Исходя из совместимости Тьюринг машины с любым языком, мы можем говорить о том, что с её помощью можно решить любую задачу, включая разработку современных веб-приложений с бизнес-логикой на стороне сервера. Проблема только в объеме работы и времени, которое потребуется для такой задачи. Поэтому трясиной будем называть решение задач с использованием инструментов требующих неадекватно большой объем времени и усилий для решения конкретной задачи.

В третьих, и самое важное, машина Тьюринга позволила математически определить понятие алгоритма, и основываясь на этой информации в дальнейшем начата разработка вопроса **вычислительной сложности**, о чем мы поговорим в следующих лекциях.

**Задачи для самостоятельного решения.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Прибавить 1 к числу:** |  | **0** | **0** | **1** | **0** | **1** | **1** | **0** | **1** |  |  |  |  |  |  |  |
| **Вычесть 1 из числа:** |  |  | **1** | **0** | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **0** |  |  |  |  |
| **Прибавит 1 к числу, если справа и слева есть другие числа** |  | **1** | **1** | **0** |  |  | **0** | **1** |  | **0** | **1** | **1** |  | **1** | **0** | **1** |
| **Прибавит ко второму числу первое** |  |  | **1** | **1** | **0** | **1** |  |  | **0** | **0** | **1** | **0** | **1** | **0** | **1** | **1** |
| **Стереть все имеющиеся значения, и добавить в другие пустые ячейки данные** | **1** | **1** |  | **0** |  | **1** | **0** |  | **0** | **1** |  |  |  | **1** |  | **0** |
| **сложить 2 числа** |  |  | **3** |  | **5** |  |  |  |  |  |  |  |  |  |  |  |

**Интернет-ресурсы:**

* Лекция 15. Машины Тьюринга. Тезис Чёрча – Тьюринга

Дискретная математика, ВШЭ, факультет компьютерных наук

(Осень 2014 – весна 2015)

<http://www.mi-ras.ru/~podolskii/files/lecture15.pdf>

* Машина Тьюринга

Материал из Википедии — свободной энциклопедии

 (27 сентября 2019)

[https://ru.wikipedia.org/wiki/Машина\_Тьюринга](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%B0_%D0%A2%D1%8C%D1%8E%D1%80%D0%B8%D0%BD%D0%B3%D0%B0)

* Аль-Хорезми

[https://ru.wikipedia.org/wiki/Аль Хорезми](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D1%8C-%D0%A5%D0%BE%D1%80%D0%B5%D0%B7%D0%BC%D0%B8)

* Эмулятор Машины Тьюринга. Краткие сведения.

Материалы по практикуму для студентов

ВМК МГУ им. М.В. Ломоносова

<http://cmcmsu.info/1course/alg.schema.mt.htm>

* Открытая информатика. Уроки 3-6. Машина Тьюринга.

Информатика и ИКТ. Профильный курс.

<https://doma10.ucoz.ru/index/mashina_tjuringa/0-87>

* Машина Тьюринга: описание и примеры МТ.

<https://www.syl.ru/article/178287/new_mashina-tyuringa-opisanie-i-primeryi-mashin-tyuringa>

**Книги:**

Чарльз Петцольд. Читаем Тьюринга // Изд-во Москва ДМК Пресс, 2014 г. – 440 с.